

ABSTRACT

A technique, system, and computer program for enhancing performance of a computer running a multithreaded server application. A scheduling heuristic is defined for optimizing the number of available threads. This heuristic alleviates over-scheduling of worker threads by defining a technique to wait to assign an incoming request to a currently-executing thread (upon completion of the thread's current work), instead of awakening a blocked thread for the incoming request. Provision is made to ensure no thread waits too long. Two stages are associated with a passive socket, so that a connection is only bound to a worker thread when work arrives for that connection. A new type of socket is defined, for merging input from more than one source and making that merged input available for scheduling. A giveback function is defined, for optimizing assignment of threads to incoming requests when persistent connections are used. Threads that go idle are put onto an idle queue, releasing them from a worker thread.